

ALGORİTMA ve AKIŞ ŞEMASI

KAZANIMLAR

- Verilen problemi çözmek üzere farklı algoritmalar tasarlar.
- Karmaşık bir problemi alt problemlere ayırarak çözümleri sözde kod olarak tasarlar..
- Algoritmayı analiz ederek sonucunu yordar.
- Verilen problemin çözümü için uygun akış şemaları oluşturur.



Ç. Volkan YILDIZ - Bilgisayar Bilimi - Ekim 2019

ALGORİTMA VE AKIŞ ŞEMASI

- Bilgisayarların çok iyi çözdüğü problemler, **algoritmik** yapıda olanlardır yani **adım adım işlemlerden oluşan** yapılardır. Bu işlemler bilgisayarın anlayabileceği biçimde ifade edilir ve bilgisayar bu satırları sıra ile çalıştırır.
- Evden okula gitmek, kütüphanede aradığın kitabı bulmak gibi günlük işlerimizde dahi algoritmaları kullanırız.

ALGORİTMA

- Herhangi bir problemin çözümüne dair adımları göstermeye **algoritma** denir.
- Bilgisayar biliminde **algoritma**, bir programın ilerleyiş adımlarını göstermek için kullanılır.
- Algoritmaları iyi bilmek, doğru şekilde ve doğru zamanda uygulamak ilginç ve önemli programlar yapmanızı sağlar.



Harezmi (780-850)

SİSTEMATİK PROBLEM ÇÖZME YAKLAŞIMI

Tüm programlar aşağıdaki işlem(lere) bağlı olarak çalışır.

- GİRDİ
- ÇIKTI
- **MATEMATİK**
- KOŞULLU YÜRÜTME
- TEKRARLAMA

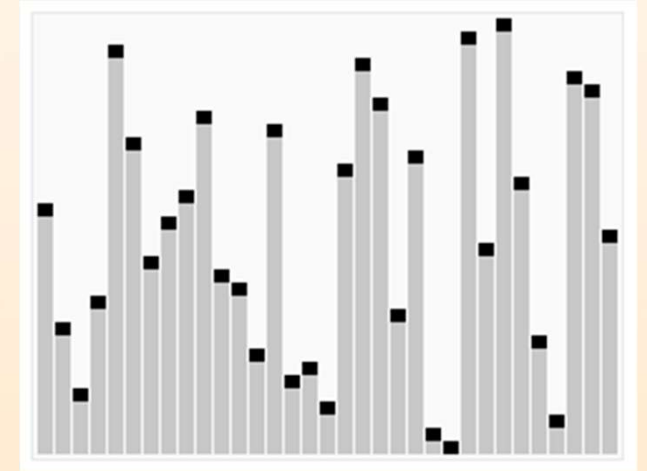
ALGORİTMA

- Basit bir örnek (*Girilen iki sayıyı toplayan algoritma*)
 1. Başla
 2. Değişkenler (toplam,s1,s2)
 3. Birinci sayıyı gir
 4. Birinci sayı = s1
 5. İkinci sayıyı gir
 6. İkinci sayı = s2
 7. toplam=s1+s2
 8. Sonucu (toplam) ekranda göster.
 9. Bitir

```
Birinci sayıyı gir:  
İkinci sayıyı gir:  
20
```

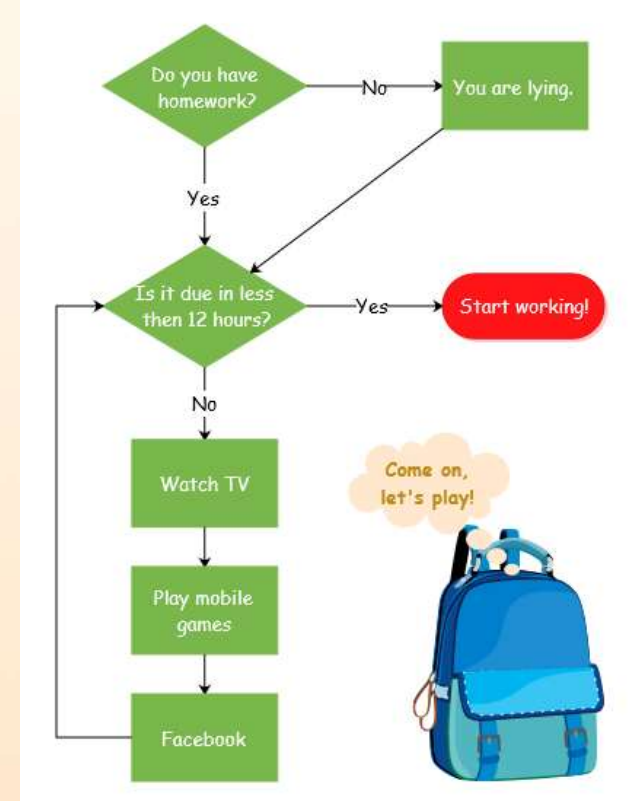
ALGORİTMA

- Daha zor bir örnek (*Hızlı sıralama algoritması*)
 1. Liste içerisinde pivot olacak bir eleman seçilir.
 2. Pivot değerden küçük olanlar pivottan önce, büyük olanlar pivottan sonra olacak şekilde elemanlar liste içerisinde yer değiştirilir.
 3. Pivotun öncesindeki ve sonrasındaki değerler ayrı bir liste kabul edilip hızlı sıralama algoritması bu listeler için yeniden çalıştırılır.



AKIŞ ŐEMASI

- Algoritmaların daha anlaşılır olabilmesi için **akış Őemaları** kullanılır.
- Akış Őemasına bakan insanlar, hangi kodları yazması gerektiğini ve algoritmanın nasıl bir Őekilde işlediğini anlamaktadırlar.



AKIŞ ŐEMASI ŐEKİLLERİ

1. Elips

- Algoritma her zaman "**Başla**" komutu ile başlar, "**Bitir**" komutu ile biter.
- "Başla" ve "Bitir"i belirtmek için "**elips**"i kullanırız

Başla

Bitir

AKIŞ ŐEMASI ŐEKİLLERİ

2. Paralelkenar

- Programımıza klavye ile **veri giriŐi** yapılacađı ya da veri iŐlendikten sonra ekranda **veri ıkıŐı** bilgisini veren semboldür.

Birinci Sayıyı
Giriniz

Not Ortalaması: 65

AKIŞ ŐEMASI ŐEKİLLERİ

3. Dikdörtgen

- **Hesaplama** veya **deęişken** ataması yapılacağını bildirir.

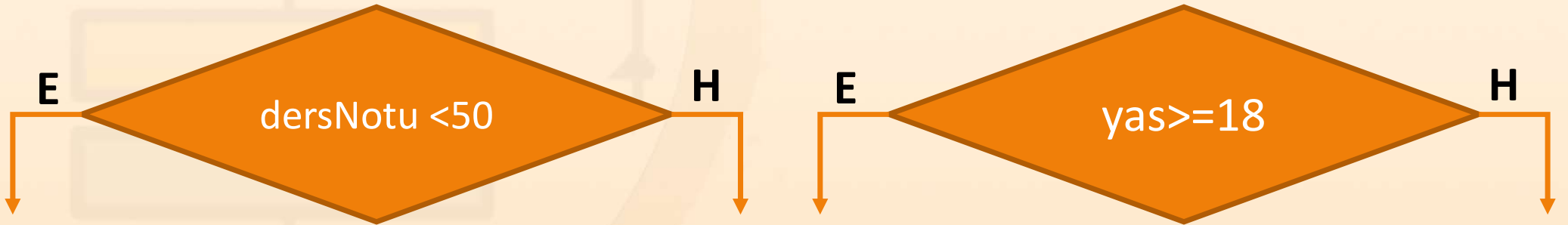
$toplam=s1+s2$

Deęişkenler
ad, okulNo

AKIŞ ŐEMASI ŐEKİLLERİ

4. EŐkenar Dörtgen

- Programda bir **karŐılaŐtırma** durumunun olduĐunu, yani programın "**Evet**" veya "**Hayır**" cevaplarına gÖre bir yol izleyeceĐini bizlere aktarır.



AKIŞ ŐEMASI ŐEKİLLERİ

5. Yön Okları

- Akış Őemasının ilerlediđi **yönü gösterir.**



AKIŞ ŐEMASI ŐEKİLLERİ

6. Altıgen

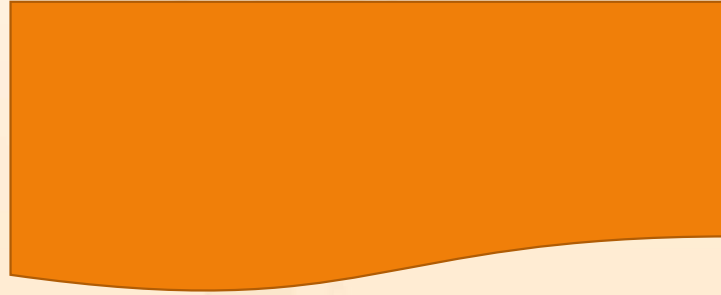
- **Tekrarlanacak** işlemler (döngü) için kullanılır.



AKIŞ ŐEMASI ŐEKİLLERİ

7. Yamuk Dörtgen

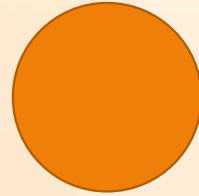
- **Ekrana yazdırmak** için kullanılır.



AKIŞ ŐEMASI ŐEKİLLERİ

8. Daire

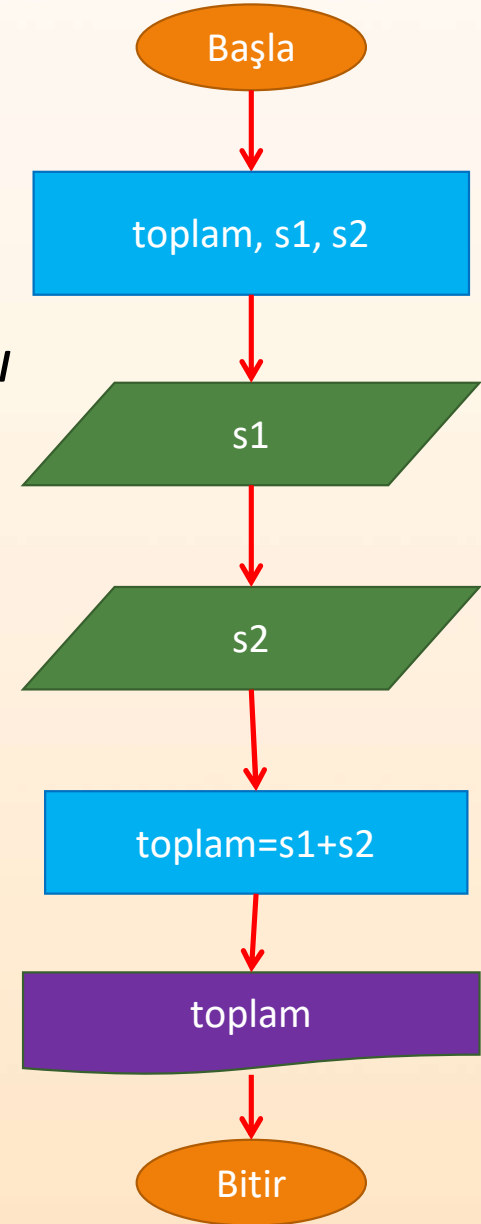
- Akıř Őemaları arasında **baęlantı kurmak** için kullanılır.



AKIŞ ŞEMASI

- *Girilen iki sayıyı toplayan algoritmanın akış şeması*

1. Başla
2. Değişkenler (toplam,s1,s2)
3. Birinci sayıyı gir
4. $s1 =$ Birinci sayı
5. İkinci sayıyı gir
6. $s2 =$ İkinci sayı
7. $toplam = s1 + s2$
8. Sonucu (toplam) ekranda göster.
9. Bitir

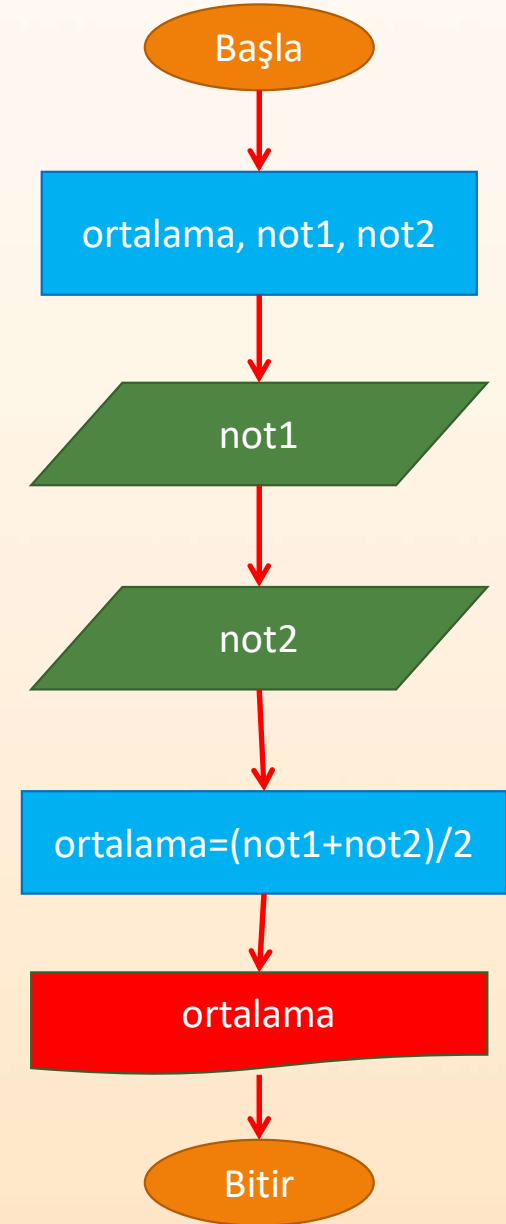


ALIŐTIRMA-1

- 2 yazılı notu olan dersin not ortalamasını ekranda gösteren akış şemasını çiziniz.
 - ✓ **Girdi** - not1 ve not2 iste
 - ✓ **Çıktı** - ortalamayı ekrana yazdır.
 - ✓ **Matematik** - not1 ve not2'yi topla ikiye böl
 - ✓ **Koşullu yürütme** - yok
 - ✓ **Tekrarlama** - yok

ALIŐTIRMA-1

1. BaŐla
2. DeęiŐkenler (ortalama,not1,not2)
3. Birinci yazılı notunu gir
4. not1 = Birinci yazılı notu
5. İkinci yazılı notunu gir
6. not2 = İkinci yazılı notu
7. $ortalama = (not1 + not2) / 2$
8. Sonucu (ortalama) ekranda göster.
9. Bitir

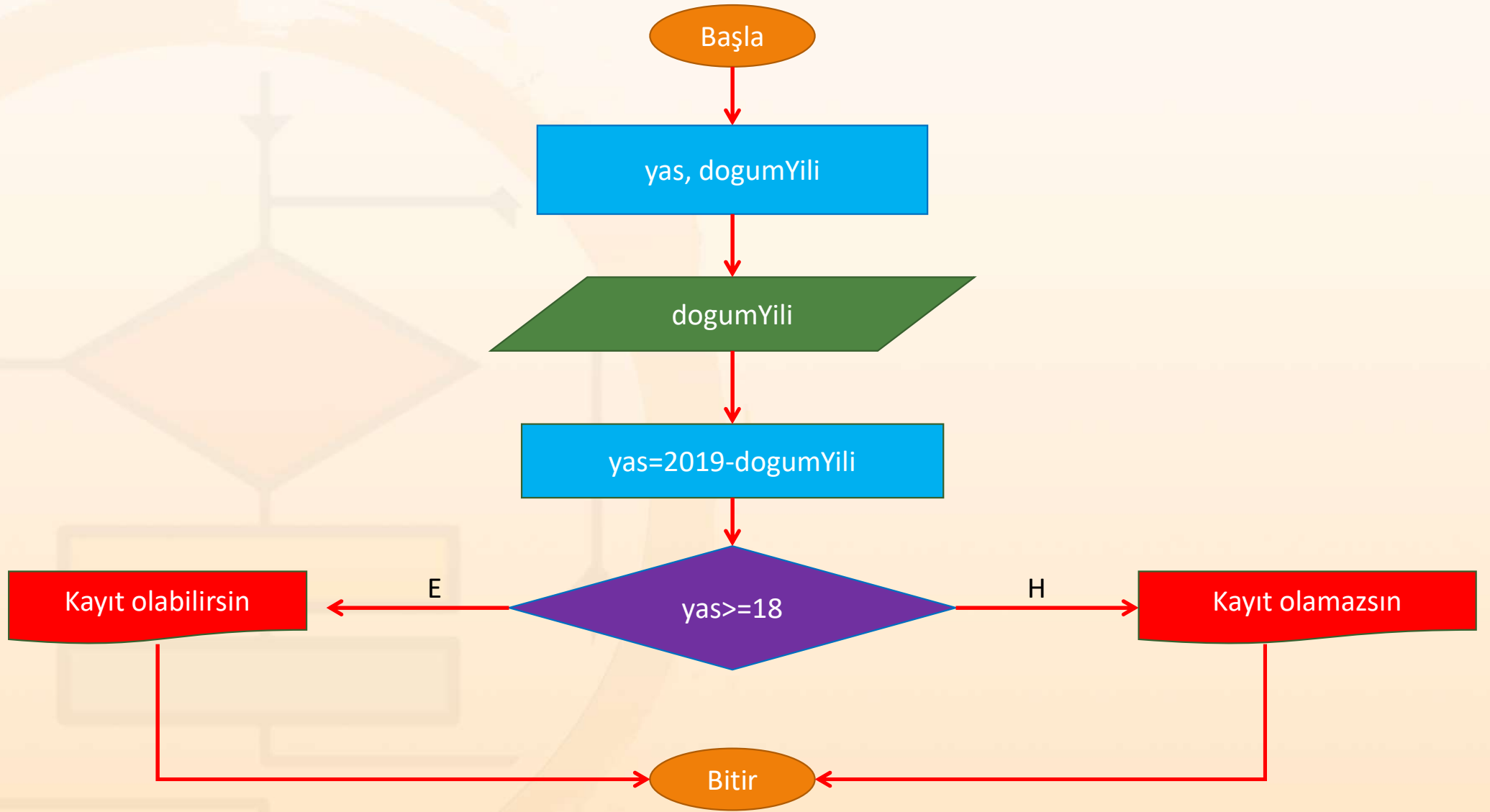


ALIŐTIRMA-2

- *Doęum yılına göre yařını bulup hesaplayan ve 18 yařından küçükse kayıt olmasına izin vermeyen akıř Őemasını çiziniz.*
 - ✓ **Girdi** - Doęum yılını iste
 - ✓ **Çıktı** - Kayıt olup olamayacağını ekrana yazdır
 - ✓ **Matematik** - Bulduğumuz yıldan doğum yılını çıkar
 - ✓ **Koşullu yürütme** - Bulunan yaş değeri 18'den küçükse kayıt olamaz, 18 veya 18'den büyükse kayıt olur.
 - ✓ **Tekrarlama** - yok

ALIŐTIRMA-2

1. BaŐla
2. DeęiŐkenler (yas, dogumYili)
3. Doęum Yılını gir
4. $dogumYili = Doęum Yılı$
5. $yas = 2019 - dogumYili$
6. YaŐı (yas) 18 veya 18'den bŸyŸkse Adım 8'e git
7. YaŐı (yas) 18'den kŸçŸk ise Adım 9'a git
8. YaŐınız 18 veya 18'den bŸyŸk kayıt olabilirsiniz.
9. YaŐınız 18'den kŸçŸk kayıt olamazsınız.
10. Bitir



ÖDEV

- Bir üniversite öğrencisinin dönem geçme notunu hesaplayıp, bu nota göre geçme kalma durumunu ekranda gösteren akış şemasını aşağıdaki şartlara göre yapınız.
 - İki vize ve bir final sınav notu vardır.
 - Dönem geçme notu, vize notlarının %40'ı, final notunun %60'ı alınıp toplanarak elde edilir.
 - Dönem geçme notu 50'dir.



KAYNAKLAR

- <https://www.elektrikport.com/teknik-kutuphane/bes-dakikada-algoritmayi-taniyin/8223#ad-image-0>
- <https://tr.khanacademy.org/computing/computer-science/algorithms/intro-to-algorithms/v/what-are-algorithms>
- <https://www.muhandisbeyinler.net/algoritma-ve-akis-semalari/>
- <https://tolpp.com/quick-sort-hizli-siralama/>
- <https://www.youtube.com/channel/UCN-0SgO68y9jyEZrJitS1nw>